

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method of communicating between ~~one~~ and a plurality of devices, comprising:
 - establishing a pseudo-terminal for ~~a device~~ each of the plurality of devices, wherein the pseudo-terminal facilitates communication between ~~the device~~ each of the plurality of devices and an application data stream;
 - receiving, from ~~[[the]]~~ a device within the plurality of devices, input to the application data stream;
 - receiving an output from the application data stream based on the received input and input from ~~[[the]]~~ other devices within the plurality of [[other]] devices; and
 - providing the output to the device and the ~~plurality of other~~ devices at substantially a same time, wherein only the output from the application data stream is shared by the device and the ~~plurality of other~~ devices.
2. (Currently amended) The method of claim 1, wherein output from the application data stream is shared by the device and the ~~plurality of other~~ devices using a data stream splitter.
3. (Original) The method of claim 2, wherein the data stream splitter is dynamically constructed to provide shared access to the application data stream.
4. (Canceled)
5. (Previously presented) The method of claim 2, wherein output received by the data stream splitter from the application data stream is sent to the pseudo-terminal and data received by the pseudo-terminal from the device is sent to the data stream splitter.
6. (Currently amended) The method of claim 2, wherein receiving input to the application data stream includes:
 - cycling through entries in a data stream splitter table to identify entries associated with the data stream splitter; and

LISTING OF THE CLAIMS:

1. (Currently amended): A method of communicating between ~~one and~~ a plurality of devices, comprising:
 - establishing a pseudo-terminal for ~~a device each of the plurality of devices~~,
 - wherein the pseudo-terminal facilitates communication between ~~the device each of the plurality of devices~~ and an application data stream;
 - receiving, from ~~[[the]]~~ a device ~~within the plurality of devices~~, input to the application data stream;
 - receiving an output from the application data stream based on the received input and input from ~~[[the]] other devices within the plurality of [[other]] devices~~; and
 - providing the output to the device and the plurality of ~~other~~ devices at substantially a same time, wherein only the output from the application data stream is shared by the device and the plurality of ~~other~~ devices.
2. (Currently amended): The method of claim 1, wherein output from the application data stream is shared by the device and the plurality of ~~other~~ devices using a data stream splitter.
3. (Original): The method of claim 2, wherein the data stream splitter is dynamically constructed to provide shared access to the application data stream.
5. (Previously presented): The method of claim 2, wherein output received by the data stream splitter from the application data stream is sent to the pseudo-terminal and data received by the pseudo-terminal from the device is sent to the data stream splitter.
6. (Currently amended): The method of claim 2, wherein receiving input to the application data stream includes:
 - cycling through entries in a data stream splitter table to identify entries associated with the data stream splitter; and

cyclically providing the device and the other devices access to the application data stream based on the cycling through the entries in the data stream splitter table.

7. (Original) The method of claim 1, further comprising:
generating a data stream splitter to handle access to the application data stream if the application data stream is not already being handled by another data stream splitter; and
adding an entry to a data stream splitter table for the device and the data stream splitter.
8. (Original) The method of claim 1, wherein the receiving input to an application data stream, receiving output from an application data stream, and the providing steps are performed by a data stream splitter manager.
9. (Currently amended) A method of communicating between ~~one~~ and a plurality of devices, comprising:
receiving, from a device within the plurality of devices, input to an application data stream;
receiving an output from the application data stream based on the received input and input from ~~[[the]]~~ other devices within the plurality of ~~[[other]]~~ devices; and
providing the output to the device and the ~~plurality of other~~ devices at substantially a same time, wherein only the output from the application data stream is shared by the device and the ~~plurality of other~~ devices;
wherein the receiving input to an application data stream, receiving output from an application data stream, and the providing steps are performed by a data stream splitter manager; and
wherein, when the data stream splitter manager receives a request for access to the application data stream from the device, the data stream splitter manager forks a copy of itself to handle the access to the application data stream for that device.
10. (Original) The method of claim 8, wherein the data stream splitter manager is transparent to a user of the device.
11. (Original) The method of claim 8, wherein the data stream splitter manager includes a graphical user interface.

12. (Original) The method of claim 1, further comprising storing data from the data stream in a buffer, wherein when the device is first provided access to the data stream, the contents of the buffer are streamed to the device.

13. (Currently amended) A method of providing ~~a device~~ shared access to a data stream for a plurality of devices, comprising:

receiving a request for access to the data stream from a device within the plurality of devices;
establishing a pseudo-terminal for ~~the device~~ each of the plurality of devices, wherein the pseudo-terminal facilitates communication between ~~the device~~ each of the plurality of devices and the data stream;

adding an entry to a data stream splitter table for the device; and

providing the device access to the data stream via a data stream splitter in accordance with the entry in the data stream splitter table, wherein providing the device access includes providing output from the data stream to the device and sending input from the device to the data stream, and wherein the output from the data stream is provided in a realtime manner based on the input from the device and input received from ~~at least one other device~~ devices within the plurality of devices.

14. (Canceled)

15. (Currently amended) A method of providing shared access to a bi-directional data stream for a plurality of devices, comprising:

establishing a pseudo-terminal for ~~a device~~ each of the plurality of devices, wherein the pseudo-terminal facilitates communication between ~~the device~~ each of the plurality of devices and the bi-directional data stream;

cycling through entries in a data stream splitter table, each entry in the data stream splitter table identifying ~~the client device~~ a device within the plurality of devices;

sending data from the bi-directional data stream to the ~~client~~ device identified in each entry based on the cycling through of the entries; and

receiving data from the ~~client~~ device identified in each entry, based on the cycling through of the entries, and sending the data from the ~~client~~ device to the bi-directional data stream.

16. (Currently amended) The method of claim 15, wherein access to the data stream is shared by [[a]] the plurality of client devices based on the entries in the data stream splitter table, each of the ~~client~~ plurality of devices having full access to the data stream.

17. (Currently amended) The method of claim 15, wherein ~~the client~~ each of the plurality of devices have a private communication channel to the data stream but the output from the data stream is shared by ~~[[all]]~~ each of the client plurality of devices.

18. (Original) The method of claim 15, wherein the sending and receiving steps are performed by a data stream splitter.

19. (Original) The method of claim 18, wherein the data stream splitter is dynamically constructed to provide shared access to the data stream.

20. (Currently amended) The method of claim 15, wherein sending data from the data stream to the client device includes sending data from the data stream splitter to the pseudo-terminal associated with the client device.

21. (Currently amended) The method of claim 15, wherein receiving data from the client device includes receiving data from the client device via the pseudo-terminal associated with the client device.

22. (Original) The method of claim 18, wherein the data stream splitter provides non-blocking raw input/output access to the data stream.

23-46. (Canceled)

47. (Currently amended) A method of communicating between ~~one and~~ a plurality of devices, comprising:

establishing a pseudo-terminal for each of the plurality of devices, wherein the pseudo-terminal facilitates communication between each of the plurality of devices and an application data stream;

receiving from at least two of the plurality of devices, input to the application data stream;

combining the input from the at least two of the plurality of devices to produce combined output;

and

simultaneously outputting the combined output at each of the plurality of devices.

48. (Currently amended) A method of communicating between ~~one~~ and a plurality of devices, comprising:

establishing a pseudo-terminal for ~~a device~~ each of the plurality of devices, wherein the pseudo-terminal facilitates communication between ~~the device~~ each of the plurality of devices and an application data stream;

receiving, from ~~[[the]]~~ a device within the plurality of devices, input to the application data stream;

receiving an output from the application data stream based on the received input and input from one or more ~~[[of]]~~ other devices within the plurality of ~~[[other]]~~ devices; and

providing the output to each of the plurality of devices at substantially a same time.

49. (Previously presented) A method of displaying an output display from an application data stream shared by a plurality of devices, comprising:

establishing a pseudo-terminal for each of the plurality of devices, wherein the pseudo-terminal facilitates communication between each of the plurality of devices and the application data stream;

receiving input from at least two of the plurality of devices;

combining the input from the at least two of the plurality of devices; and

displaying, substantially simultaneously, an output display based on the combined input from the at least two of the plurality of devices at the at least two of the plurality of devices.